

Report – Print Job Scheduling with Priority Queue Using Max-Heap

Program's functionality:

The program implements a print job scheduling system that manages print jobs based on priority using a max-heap structure. The main functionality includes:

- 1. Inserting Print Jobs:** Users can add print jobs to the queue, each with a unique name and their corresponding priority. Higher-priority jobs are processed before lower-priority jobs.
- 2. Processing Print Jobs:** The job with the highest priority is always processed first, and after processing, the max-heap properties are restored.
- 3. Updating Job Priority:** Users can update the priority of an existing job, and the program adjusts the job's position in the heap to ensure that the highest-priority job remains at the top.
- 4. Error handling:** The program handles cases such as duplicate job names, invalid priority inputs, and attempts to update non-existent jobs.

The user-friendly interface gives users the option to enter job details, view the next job in line, process jobs, update priority of a job, view all jobs sorted by priority, and directly exit the program.

Examples:

Scenario: At the university's single, overloaded printer, students are impatient to get their print jobs completed. They know that updating their job's priority is the key to getting ahead in the print queue.

Program interaction flow [Output/Input]:

Print Job Scheduling System

Choose one of the options:

1. Insert print job
2. Display next print job (print job with highest priority)
3. Process next print job
4. Update print job priority
5. Display all print jobs
6. Exit program

Your choice:1

Enter job name (only single words are allowed):ResearchPaper

Enter job priority:1

Job "ResearchPaper" successfully added.
Highest priority: ResearchPaper (Priority: 1)

Print Job Scheduling System

Choose one of the options:

1. Insert print job
2. Display next print job (print job with highest priority)
3. Process next print job
4. Update print job priority
5. Display all print jobs
6. Exit program

Your choice:1

Enter job name (only single words are allowed):LabReport

Enter job priority:2

Job "LabReport" successfully added.
Highest priority: LabReport (Priority: 2)

Print Job Scheduling System

Choose one of the options:

1. Insert print job
2. Display next print job (print job with highest priority)
3. Process next print job
4. Update print job priority
5. Display all print jobs
6. Exit program

Your choice:2

LabReport (Priority: 2)

Print Job Scheduling System

Choose one of the options:

1. Insert print job
2. Display next print job (print job with highest priority)
3. Process next print job
4. Update print job priority
5. Display all print jobs
6. Exit program

Your choice:4

Enter name of job you want to update: ResearchPaper

Enter new priority: 3

Priority of "ResearchPaper" is updated to 3.

Jobs in priority order (highest to lowest):

Job name: ResearchPaper, Job priority: 3

Job name: LabReport, Job priority: 2

Print Job Scheduling System

Choose one of the options:

1. Insert print job
2. Display next print job (print job with highest priority)
3. Process next print job
4. Update print job priority
5. Display all print jobs
6. Exit program

Your choice: 2

ResearchPaper (Priority: 3)

Print Job Scheduling System

Choose one of the options:

1. Insert print job
2. Display next print job (print job with highest priority)
3. Process next print job
4. Update print job priority
5. Display all print jobs
6. Exit program

Your choice: 3

Printing job: ResearchPaper (Priority: 3)

Print Job Scheduling System

Choose one of the options:

1. Insert print job
2. Display next print job (print job with highest priority)
3. Process next print job
4. Update print job priority
5. Display all print jobs
6. Exit program

Your choice:5

Jobs in priority order (highest to lowest):

Job name: LabReport, Job priority: 2

Print Job Scheduling System

Choose one of the options:

1. Insert print job
2. Display next print job (print job with highest priority)
3. Process next print job
4. Update print job priority
5. Display all print jobs
6. Exit program

Your choice:6

Exiting program..

Testing process and results:

I tested the program using the example scenario given in the assignment, which involved adding jobs with varying priorities, updating job priorities, displaying next job, and processing jobs. The program's output matched the expected results provided in the example, confirming that the core functionality works as required.

In addition to verifying the core functionality, I tested the program's handling of unexpected inputs to ensure handling of errors, and a user-friendly program:

Duplicate job names: I attempted to add two jobs with the same name. The program correctly identified the duplicate and displayed an error message, preventing duplicate entries in the queue.

Invalid data types for choices and priority: When prompted to enter choice or priority, I entered a string instead of a number. The program identified the invalid input and prompted me to re-enter a valid integer.

Sentence as job names: I tested entering a sentence as a job name. The program accepted only the first word as the job name and discarded the remaining text, keeping names consistent with the example in the assignment.

No input given: If I left the input field blank, the program simply waited without proceeding.

I tested the program with other edge cases, including but not limited to very high/low priority values, processing from an empty queue, attempting to update a non-existent job, and non-integers input for choices and priority. The program handles all of these scenarios.